An Evolutionary Algorithm for the Electric Vehicle Routing Problem with Battery Degradation and Capacitated Charging Stations

Juan Pablo Futalef[1], Diego Muñoz-Carpintero[2], Heraldo Rozas[1], Marcos Orchard [1]

[1] *Department of Electrical Engineering, University of Chile, Santiago, Chile*
juan.futalef@ing.uchile.cl
heraldo.rozas@ing.uchile.cl
morchard@ing.uchile.cl
[2] *Institute of Engineering Science, Universidad de O'Higgins, Rancagua, Chile*
diego.munoz@uoh.cl

Abstract

As CO2 emission regulations increase, fleet owners increasingly consider the adoption of Electric Vehicle (EV) fleets in their business. The conventional Vehicle Routing Problem (VRP) aims to find a set of routes to reduce operational costs. However, route planning of EVs poses different challenges than that of Internal Combustion Engine Vehicles (ICEV). The Electric Vehicle Routing Problem (E-VRP) must take into consideration EV limitations such as short driving range, high charging time, poor charging infrastructure, and battery degradation. In this work, the E-VRP is formulated as a Prognostic Decision-Making problem. It considers customer time windows, partial midtour recharging operations, non-linear charging functions, and limited Charge Station (CS) capacities. Besides, battery State of Health (SOH) policies are included in the E-VRP to prevent early degradation of EV batteries. An optimization problem is formulated with the above considerations, when each EV has a set of costumers assigned, which is solved by a Genetic Algorithm (GA) approach. This GA has been suitably designed to decide the order of customers to visit, when and how much to recharge, and when to begin the operation. A simulation study is conducted to test GA performance with fleets and networks of different sizes. Results show that E-VRP effectively enables operation of the fleet, satisfying all operational constraints.

1. Introduction

Fleet owners continuously strive to operate their fleets efficiently. In economic affairs, fleet owners attempt to minimize operational costs while ensuring operational constraints. Nonetheless, the minimization of operational costs does not imply an environmental-friendly operation (Dekker, Bloemhof, & Mallidis, 2012). In the case of land vehicles, their operation phase contributes to the most Greenhouse Gasses (GHG) emission of their whole life cycle (Faria et al., 2013). Moreover, they contribute to significant space usage and noise pollution (Lee & Sener, 2016).

The well-known Vehicle Routing Problem (VRP) arises as an intuitive optimization problem to look for efficient routes. The VRP aims at finding the least-cost routes a set of vehicles can follow to serve a finite number of customers (Kumar & Panneerselvam, 2012)(Cattaruzza, Absi, Feillet, & González-Feliu, 2017). The routes are constrained to the nature of the operation to execute. The VRP is an NP-hard problem, i.e. the computational effort required to solve it grows exponentially when the problem size increases. Consequently, exact solutions of large real-world problems become computationally intractable quickly, even when only a few hundred customers are considered (Nazari, Oroojlooy, Snyder, & Takác, 2018). Despite this issue, fleet owners require solving big problem instances in short periods. This hard requirement makes exact algorithms unpractical; thus, researchers and commercial solvers have focused on metaheuristics, which are generic heuristics dedicated to efficiently explore the search space so as to determine near–optimal solutions. Some of these metaheuristics include: Genetic Algorithms (GA), Ant Colony Optimization (ACO), Tabu Search (TS), Simulated Annealing (SA), among others (Gendreau, Potvin, Bräumlaysy, Hasle, & Løkketangen, 2008).

The Electric Vehicle Routing Problem (E-VRP) is a VRP variant where all vehicles are Electric Vehicles (EV) (Montoya, Guéret, Mendoza, & Villegas, 2017) (Erdelic & Caric, 2019). We consider Battery Electric Vehicles (BEV), which are only powered from batteries installed inside the vehicles. Therefore, many advantages and disadvantages arise. Some of the most critical BEV advantages include: they are much quieter than Internal Combustion Engine Vehicles (ICEV), they do not emit GHG locally, and there is the possibility to recharge batteries from clean sources (Faria et al., 2013)(Notter et al., 2010). Most BEV disadvantages derive from the limited battery capacity and the little recharging infrastructure available. Battery capacity directly impacts the driving range of a BEV, which averages around 200 km, nearly one-third the average driving range of an ICEV (Erdelic & Caric, 2019)(Feng & Figliozzi, 2013). This limitation forces BEVs to visit charging stations (CS) frequently. However, most cities around the world are not equipped with large and robust recharging infrastructure yet. Consequently, a properly formulated E-VRP must address both BEV constraints and recharging infrastructure limitations.

Another significant BEV limitation comes from battery degradation. Studies have shown that improper handling of batteries can lead to an increase in long-term operational costs (Perez, Moreno, Moreira, Orchard, & Strbac, 2016). This cost increase occurs because mishandled batteries degrade faster, which forces fleet owners to replace them early. Furthermore, this replacement produces a large environmental burden: GHG emissions from battery manufacturing are comparable to manufacturing all the rest of the vehicle itself (Faria et al., 2013) (Notter et al., 2010).

To increment the battery lifespan, one can constrain the State of Charge (SOC) of the battery (Perez et al., 2016). We denominate this method as State Of Health (SOH) policy. Some studies consider this kind of policy into the E-VRP formulation. In (Barco, Guerra, Muñoz, & Quijano, 2017), they address an airport shuttle with BEVs, and directly constrain SOC to prevent the battery from degrading faster. In (Sassi, Cherif, & Oulamara, 2014), they tackle the delivery of goods with a mixed fleet of ICEV and BEV, while incorporating the same SOH policy. However, the latter two works do not mention CS capacity, which, according to (Froger, Mendoza, Jabali, & Laporte, 2017), influences the feasibility of solutions.

In this work, we propose a new formulation of the E-VRP considering non-linear charging functions, battery degradation, and limited CS capacity. To solve the problem, we develop a custom Genetic Algorithm (GA). This GA is capable of inserting charging operation on-execution; thus, no additional methodologies and heuristics are required. Results show that the GA is capable of finding optimal feasible routes each vehicle can follow.

The paper structure is as follows. Section 2 describes the problem in detail. Section 4 introduces the GA to solve the problem. Section 5 shows results of four experiments of different complexity. Finally, section 6 concludes about the work and future trends to improve research.

2. THE E-VRP-NL-TW WITH BATTERY DEGRADATION

2.1. Problem Statement

Consider a directed network $D = (\{0\} \cup N \cup F, A)$, where $\{0\}$ is the depot, $N = \{1, \dots, n\}$ are n nodes representing customers to be visited, and $F = \{n+1, \dots, n+l\}$ are l nodes representing all available CS. The set V represents all the nodes in the network, i.e. $V = N \cup \{0\} \cup F$. The set $A = \{(i,j) : i, j \in N \cup \{0\} \cup F, i \neq j\}$ are all the arcs connecting the network nodes.

Each customer $j \in N$ requires a demand D_j, which is delivered by an EV. When the EV arrives at j, it takes a known time T_j until the EV departs from j. A time window $[T_j^-, T_j^+]$ constrains this service, i.e., the EV must arrive after T_j^- and leave before T_j^+. Both T_j^- and T_j^+ belong to the set $\Omega \subset \mathbb{R}^+$, which contains the times of the day. The time window is such that its width is larger than the service time.

In order to serve all customers, a fleet M of m EVs is available. Each EV $i \in M$ must visit a set of previously assigned customers N^i. This assignation of customers per vehicle is always the same throughout operation, and it is such that the EV does not carry more weight than its weight limit. Additional to its weight limit, each EV i has a battery capacity $\bar{Q}_i$. All EVs have the same maximum tour time duration of $\bar{T}$.

Travel across arc $(i,j) \in A$ defines the travel time the EV must spend, and energy consumption the EV battery spends. The travel time is t_{ij}, while the energy consumption is

$$e_{ij}(w) = \bar{e}_{ij} \times (w + w_{EV}),$$

where $\bar{e}_{ij}$ is the per-weight-unit energy cost, w is the weight the EV carries across the arc, and w_{EV} is the EV weight. There are two methods to calculate both t_{ij} and $\bar{e}_{ij}$: first, assume they are a function of the Euclidean distance among i and j; second, solve the Shortest Path Problem (SPP), which aims to find the road path that minimizes the distance among two nodes.

The SOC is a representation of the remaining energy in the battery (Pola et al., 2015). When the battery is full, the SOC is one; on the other hand, when the battery is empty, the SOC is zero. Several studies tackle the E-VRP by assuming SOC instead of energy consumption because it does not rely on EV parameters(Montoya et al., 2017)(Sassi et al., 2014). Following the same assumption, we will assume that e_{ij} is the decrease in the battery SOC when the EV travels from i to j.

When the EV battery operates with a SOC too close to zero or one for long periods, it tends to degrade faster (Perez et al., 2016). Therefore, to prevent early degradation of EV batteries, we consider a State Of Health (SOH) policy. This policy consists of constraining the battery SOC in an interval $[\alpha^-, \alpha^+]$, where $\alpha^-, \alpha^+ \in [0, 100]\%$.

During operation, vehicles can visit any CS to recharge the battery. Each CS $j \in F$ can do a limited number of r_j parallel charging operations. A charging operation made at CS j takes time $\Delta_j(p, q)$, where p is the SOC when the EV reaches the CS, and q is the SOC increase made by the charging operation. This operation is characterized by a charging function, which is concave and non-linear (Montoya et al., 2017). To calculate charging times, we consider the method presented by (Montoya et al., 2017), where a piece-wise function approximates the charging function of a CS.

The criteria to choose the routes is to minimize total travel times, total energy consumption, and total charging times. A solution of the problem must return: an ordered sequence of nodes each EV will visit, the time of the day each EV departs from the depot, and, if visiting a CS, the SOC increase made by the charging operation. A feasible solution satisfies:

- all customers are visited precisely once,
- all EVs do not exceed their limitations,
- CS capacities are not surpassed,
- all EVs begin and end operation at the depot,
- all customers are visited within their time windows.

2.2. Electric Vehicle State-Space Model

The path EV $i \in M$ will follow is

$$S^i = [S_0^i, \dots, S_{s_i-1}^i]^T$$

where s_i is the length of S^i. Each S_k^i is the k-th node the EV visits. As the number of customers each EV is serving is fixed, the length s_i varies if charging operations are inserted. The vector

$$L^i = [L_0^i, \dots, L_{s_i-1}^i]^T$$

defines the charging plan, where L_k^i represents the SOC increase made in the k-th stop. Thus, if the node to visit is a customer node, $L_k^i = 0$. On the other hand, if the node to visit is a CS, there must be a SOC increase, i.e., $L_k^i > 0$.

EV i state variables are three:

- $x_1^i(k)$ is the time of the day the vehicle arrives at the k-th stop. The arrival time at stop $k+1$ depends on the arrival time at stop k plus: the time the EV spends at stop k, and the travel time from stop k to stop $k+1$. Initial condition $x_1^i(0)$ is the time of the day the vehicle leaves the depot;
- $x_2^i(k)$ is the battery SOC when the vehicle arrives at k-th stop. When the vehicle arrives at $k+1$ stop, the battery SOC depends on SOC at the previous stop k minus the SOC decrease by traveling from stop k to stop $k+1$. The SOC decrease depends on the weight the EV carries across the arc, with is tracked by state variable $x_3(k)$ If the EV visits a CS at instant k, then the SOC at $k+1$ must add this SOC increase. Initial condition $x_2^i(0)$ is the battery SOC when the vehicle leaves the depot;
- $x_3^i(k)$ is the weight the vehicle carries when it arrives to stop k. Thus, the weight at instant $k+1$ is the weight at instant k minus the requirement at stop k. If stop k is a CS, then the requirement is zero. Initial condition $x_3^i(0)$ is the sum of all customer demands to be delivered by the vehicle plus the EV weight.

The state equations use a discrete time k that represents when the EV arrives to the k-th stop in its path. Consequently, each vehicle has its own time k, which ranges from 0 to $s_i - 1$. Equations (1) to (3) are case-wise defined. These cases depend on the node type the EV visits: a customer or a CS.

$$x_1^i(k+1) = \begin{cases} x_1^i(k) + T_{S_k^i} + t_{S_k^i S_{k+1}^i} & \text{if} \quad S_k^i \in N \\ x_1^i(k) + \Delta_{S_k^i}(x_2(k), L_k^i) & \text{if} \quad S_k^i \in F \\ \quad +t_{S_k^i S_{k+1}^i} & \end{cases} \tag{1}$$

$$x_2^i(k+1) = \begin{cases} x_2^i(k) - e_{S_k^i S_{k+1}^i}(x_3^i(k)) & \text{if} \quad S_k^i \in N \\ x_2^i(k) - e_{S_k^i S_{k+1}^i}(x_3^i(k)) & \text{if} \quad S_k^i \in F \\ \quad +L_k^i & \end{cases} \tag{2}$$

$$x_3^i(k+1) = \begin{cases} x_3^i(k) - D_{S_k^i} & \text{if} \quad S_k^i \in N \\ x_3^i(k) & \text{if} \quad S_k^i \in F \end{cases} \tag{3}$$

where $T_{S_k^i}$ is the customer service time at node S_k^i, $\Delta_{S_k^i}(x_2(k), L_k^i)$ is the charging time at CS S_k^i when the EV arrives with SOC $x_2(k)$ and leaves with SOC $x_2(k) + L_k^i$; t_{ij} and $e_{ij}(w)$ are the travel time and SOC decrease when the EV travels from node i towards node j; $D_{S_{k+1}^i}$ is the demand requirement at node S_{k+1}^i.

A good forecasting of the EV state variables is essential to provide a high-quality solution to the problem. In our case, SOC prognosis is crucial to obtain solutions that satisfy the SOH policy. The incorporation of the EV weight and the per-weight energy consumption allows us to predict the SOC decrease with great detail, which is determinant on E-VRP research, and there is still a lack of E-VRP variants tackling this issue (Erdelic & Caric, 2019).

To simplify notation, Eqs. (1) to (3) are redefined as:

$$x_1^i(k+1) = F_1\left(x_1^i(k), S_k^i, S_{k+1}^i, L_k^i\right) \tag{4}$$

$$x_2^i(k+1) = F_2\left(x_2^i(k), S_k^i, S_{k+1}^i, L_k^i\right) \tag{5}$$

$$x_3^i(k+1) = F_3\left(x_3^i(k), S_k^i, S_{k+1}^i, L_k^i\right) \tag{6}$$

2.3. Counting the Number of Vehicles at Each Node

The operation of different EVs is coupled by the constraint of CS capacity, as the presence of an EV in a charging station reduces the capacity for the remaining EVs. In order to enforce this constraint, a method to count the number of vehicles at each node at any time is needed. The purpose of this section is to present this method, which cannot be done with the standard EV state variables.

In this case, a different discrete time counter is needed. A global discrete time counter $\bar{k}$ is used, which tracks the occurence of two events: an EV arriving at a node and an EV leaving from a node.

To count the number of vehicles, consider the vector $\theta(\bar{k}) \in \mathbb{R}^{|V|}$. This vector contains as many components as nodes in the network. The j-th component of $\theta(\bar{k})$ contains the number of vehicles at node j at the moment the $\bar{k}$-th event is triggered. For example, if at instant $\bar{k}$ two vehicles are visiting node j, then $\theta_j(\bar{k}) = 2$. If any of the two vehicles leave j, the event $\bar{k}+1$ "leaving a node" is triggered, and the value $\theta_j(\bar{k}+1) = 1$. This implies that the counting vector evolves every time the counter $\bar{k}$ changes as follows:

$$\theta(\bar{k}+1) = \theta(\bar{k}) + \delta(\bar{k})\gamma(\bar{k}) \tag{7}$$

where $\gamma(\bar{k}) \in \mathbb{R}^{|V|}$ is a vector such that its j-th component satisfies

$$\gamma_j(\bar{k}) = \begin{cases} 1 & \text{if} \quad \text{at instant } \bar{k}\text{, event occurs at node } j \\ 0 & \quad\quad \text{otherwise,} \end{cases} \tag{8}$$

and $\delta(\bar{k})$ is such that

$$\delta(\bar{k}) = \begin{cases} +1 & \text{if} \quad \text{at instant } \bar{k}\text{, a vehicle arrives to a node} \\ -1 & \text{if} \quad \text{at instant } \bar{k}\text{, a vehicle leaves a node.} \end{cases} \tag{9}$$

The vector $\gamma(\bar{k})$ tracks the node where the event happens, and $\delta(\bar{k})$ tracks the kind of event. The initial condition of $\theta(\bar{k})$ is $\theta(0) = [m, 0, \ldots, 0]^T$, since all m vehicles are at depot at the start of operation.

3. NONLINEAR PROGRAMMING FORMULATION

The problem is formulated as a non-linear program. Decision variables are node sequences S^i, SOC increase sequences L^i, and departure times x_0^i for each EV $i \in M$. The cost is the weighted sum of three elements: total travel times among nodes; total charging times; and total energy consumption:

$$\begin{aligned} J(S, L, x_0) = \quad & \omega_1 \underbrace{\sum_{i=1}^{m} \sum_{k=0}^{s_i - 1} t_{S_k^i S_{k+1}^i}}_{\text{Total travel time}} \\ & + \omega_2 \underbrace{\sum_{i=1}^{m} \sum_{\substack{S_k^i \in F \\ k \in K^i}} \Delta_{S_k^i}(x_2^i(k), L_k^i)}_{\text{Total charging time}} \\ & + \omega_3 \underbrace{\sum_{i=1}^{m} \sum_{k=0}^{s_i - 1} e_{S_k^i S_{k+1}^i}\left(x_3^i(k)\right)}_{\text{Total energy consumption}} \end{aligned} \tag{10}$$

where $S = [S^1, \ldots, S^m]$ $L = [L^1, \ldots, L^m]$, and $x_0 = [x_0^0, \ldots, x_0^m]$ are the decision variables. The weights ω_1, ω_2, and ω_3 tune the importance given to the item they multiply.

The following constraints are initial conditions. Equation (11) forces all EV to start from depot; Eq. (12) indicates that EVs do not increase their SOC at the beginning of the tour; Eq. (13) forces each EV to have their own starting time; Eq. (14) implies that each EV leaves the depot with the maximum value allowed by the SOH policy; Eq. (15) implies that each EV leaves the depot carrying a weight equal to the sum of the demands of all customers the EV must visit.

$$S_0^i = 0 \qquad \forall i \in M \tag{11}$$

$$L_0^i = 0 \qquad \forall i \in M \tag{12}$$

$$x_1^i(0) = x_0^i \qquad \forall i \in M \tag{13}$$

$$x_2^i(0) = \alpha^+ \qquad \forall i \in M \tag{14}$$

$$x_3^i(0) = \sum_{S_k^i \in N} D_{S_k^i} + w_{EV} \qquad \begin{aligned} &\forall i \in M \\ &\forall k \in K^i \end{aligned} \tag{15}$$

The following constraints are terminal conditions. Equation (16) forces all EVs to end at depot; Eq. (17) prevents EVs tour duration to exceed the maximum tour time.

$$S_{s_i - 1}^i = 0 \qquad \forall i \in M \tag{16}$$

$$x_1^i(s_i - 1) \leq \bar{T} + x_1^i(0) \quad \forall i \in M \tag{17}$$

Equation (18) prevents EVs from carrying more weight than their maximum weight limitation.

$$\sum_{S_k^i \in N} D_{S_k^i} \leq \bar{D}_i \quad \begin{aligned} &\forall i \in M \\ &\forall k \in K^i \end{aligned} \tag{18}$$

Equations (19) and (20) define time windows.

$$T_{S_k^i}^- \leq x_1^i(k) \qquad \begin{aligned} &\forall i \in M, \\ &\forall k \in K^i, \\ &S_k^i \in N \end{aligned} \tag{19}$$

$$x_1^i(k) \leq T_{S_k^i}^+ - T_{S_k^i} \quad \begin{aligned} &\forall i \in M, \\ &\forall k \in K^i, \\ &S_k^i \in N \end{aligned} \tag{20}$$

Equations (21) to (23) are state equations of each EV.

$$x_1^i(k+1) = F_1\left(x_1^i(k), S_k^i, S_{k+1}^i, L_k^i\right) \quad \begin{aligned} &\forall i \in M \\ &\forall k \in K^i \end{aligned} \tag{21}$$

$$x_2^i(k+1) = F_2\left(x_2^i(k), S_k^i, S_{k+1}^i, L_k^i\right) \quad \begin{aligned} &\forall i \in M \\ &\forall k \in K^i \end{aligned} \tag{22}$$

$$x_3^i(k+1) = F_3\left(x_3^i(k), S_k^i, S_{k+1}^i, L_k^i\right) \quad \begin{aligned} &\forall i \in M \\ &\forall k \in K^i \end{aligned} \tag{23}$$

Equation (24) are the dynamics of the counting vector and Eq. (25) the initial value of the counting vector. Equation (26) limit the maximum parallel charging operations at CSs.

$$\theta(\bar{k}+1) = \theta(\bar{k}) + \delta(\bar{k})\hat{\gamma}(\bar{k}) \quad \forall \bar{k} \in K^0 \tag{24}$$

$$\theta(0) = [m, 0, \ldots, 0]^T \tag{25}$$

$$\theta_j(\bar{k}) \le r_j \qquad \forall j \in F, \ \forall \bar{k} \in K^0 \tag{26}$$

The following constraints are SOH policies. Equation (27) bounds SOC of vehicles arriving at a node, while Eq. (28) bounds SOC of vehicles leaving from a node.

$$\alpha^- \le x_2^i(k) \le \alpha^+ \qquad \forall i \in M \ \forall k \in K^i \tag{27}$$

$$\alpha^- \le x_2^i(k) + L_k^i \le \alpha^+ \qquad \forall i \in M \ \forall k \in K^i \tag{28}$$

The following constraints define decision variables domains.

$$S_k^i \in N^i \cup F \quad L_k^i \in \mathbb{R}_0^+ \quad x_0^i \in \Omega \quad \forall i \in M, \ \forall k \in K^i \tag{29}$$

Finally, the problem is to find a set of node sequences S^*, a set of charging plans L^*, and a set of departure times x_0^* such that the cost function in Eq. (10) is minimized. These optimal decision variables are such that all above constraints are satisfied. Equation (30) summarizes the problem as an optimization problem.

$$S^*, L^*, x_0^* = \begin{array}{c} \arg\min\limits_{S,L,x_0} J(S, L, x_0) \\ \text{s.t.} \\ \text{Eqs. (11) - (29)} \end{array} \tag{30}$$

4. GENETIC ALGORITHM TO SOLVE THE E-VRP-TW

The optimization problem of Eq. (30) is nonlinear, and contains continuous and integer (combinatorial) variables; thus, it cannot be solved in polynomial time. Because of this, a Genetic Algorithm (GA) (Goldberg, 1989) is proposed for solving the newly proposed E-VRP because of its capacity to handle this type of problem. GA is a population-based optimization scheme, that mimics natural evolution in a population. Several operators inspired on the evolutionary process (mutation, crossover, and selection) that are used over several generations in the algorithm, are designed to drive a population of individuals to find or approximate solutions to the optimum. Mutation and crossover operations aid the diversity of the population, which helps solving problems with multiple minima, and selection drives the population to the best solutions. A pseudo-code describing a generic GA is presented in Algorithm 1.

In this section we define each of the used operators, including the ad-hoc encoding and mutation operator, defined specifically for this formulation.

Algorithm 1: Generic GA

```
$k \leftarrow 0$;
*P(k)* $\leftarrow$ Initial population;
*evaluate(P(k))*;
**while** *notFinished()* **do**
    $P_0 \leftarrow$ *select(P(k))*;
    $P_1 \leftarrow$ *crossover*($P_0$);
    *mutate*($P_1$);
    *evaluate*($P_1$);
    $P(k+1) \leftarrow$ *newPopulation*$(P_1, P(k))$;
    $k \leftarrow k + 1$;
**end**
**return** *bestIndividual(P(k))*
```

4.1. Encoding Routes

We consider a hybrid strategy of direct and indirect representation of routes in the individuals. The whole individual is the concatenation of several sub-individuals, as shown in Figure 1; each one of these sub-individuals stores the route information of a single EV.

Three blocks form a sub-individual: a customer block, a charging operation block, and a departure time block. For EV i, a customer block is a vector of integers of length equal to the number of customers in N^i which defines the sequence the customers will be visited. The k-th component of this vector is the customer ID of the k-th customer the vehicle will visit in its sequence. Notice this does not define the route of the EV because this does not consider charging operations.

A charging operations block defines in-route charging operations. The whole block is the concatenation of several charging operations sub-blocks, as shown in Figure 2. A sub-block structure is interpreted as follows. After the EV serves customer N_j^*, it goes to CS F_j^* and charges the amount q_j. If N^*j is -1, then the charging operation does not take place at all. Each EV is allowed to make n^* charging operations at most. Consequently, the whole charging operations block is made up of n^* charging operation sub-blocks. A departure time block is a single real number denoting the time of the day the EV will leave the depot.

The operations $\Psi^S(I_j)$, $\Psi^L(I_j)$ and $\Psi^{x_0}(I_j)$ are decoding operations that convert the individual I_j into decision variables S, L and x_0, respectively. The operation $\Psi(I_j)$ returns the optimization vector as a result of the individual I_j.

4.2. Genetic Operators

4.2.1. Crossover

The crossover operator creates two children by recombination of two parents. In this work, the crossover first chooses an EV and a block associated with this EV; the blocks can be the customer block, the charging operation block or the departure time block. Then, for two randomly selected parents,

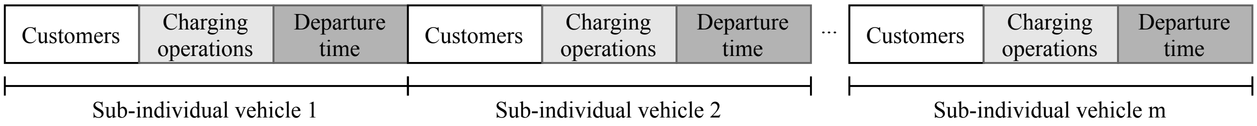


Figure 1. A full individual is made up of the concatenation of each vehicle sub-individual

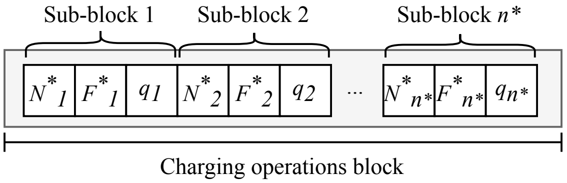


Figure 2. A full charging operations block is the concatenation of n^* charging operation sub-blocks.

the operator generates two children by copying the parents with the chosen blocks swapped.

4.2.2. Mutation

The mutation operator modifies a randomly chosen block of a single individual. The modification depends on the block type:

- Customer block: the mutation operator will randomly swap two customers in their positions.
- Charging operations block: the mutation operator modifies a single randomly-chosen sub-block. The following procedure is applied: the customer is replaced by a value randomly sampled from $N^i \cup \{-1\}$, the CS is replaced by a value randomly sampled from F, and the charging amount is modified by a random number.
- Departure time block: the value is modified by adding a random number sampled from the interval $[-60, 60)$.

4.2.3. Selection

In this work, a tournament selection of size τ is considered. Under this approach, the best individual might be lost; thus, GA considers elitism as well. We implement elitism by preserving the best κ individuals of generation k, and appending them into generation $k+1$. The worst κ individuals of generation $k+1$ are dismissed.

4.3. Constraint Handling

In this work, inequality constraints are handled by a penalization scheme. In some constraints a large negative constant is added to the fitness when the individual is outside the feasible zone. In other constraints, a value dependent on the distance to the feasible zone is added to the fitness. The distance function is

$$D(I_j) = \sum_{i \in \Omega'} \left(f_i(\Psi(I_j)) - b_i\right)^2 \tag{31}$$

where I_j is the j-th individual in the population, $\Psi(I_j)$ is the optimization vector after decoding I_j, and $f_i(\Psi(I_j))$ and b_i are such that constraint $f_i(\Psi(I_j)) \leq b_i$ cannot be satisfied, i.e. are outside the feasible zone Ω.

This quadratic penalization scheme allows the GA to violate inequality constraints by small amounts. We permit this behavior because most of these constraints are soft constraints, as it is possible that EVs reach customers outside their time windows or operate the battery outside the SOH policy by little amounts.

Equality constraints are hard-constraints; thus, the penalty function does not make use of them. Instead, equality constraints are input-evaluated via recursion. Therefore, the GA requires the definition of the one-step prediction, the initial conditions, and the sequence of future inputs. All of these requirements are obtained by decoding the individual.

4.4. Fitness

For feasible individuals, the fitness function Γ is minus the cost function in Eq. (10). For infeasible individuals, the fitness function adds a penalization equal to minus the distance function in Eq. (31) minus a large positive number C. The above two cases are summarized in the following case-wise function:

$$\Gamma(I_j) = \begin{cases} -J_{cost} & \text{if} \quad \text{feasible,} \\ -J_{cost} - D\left(\Psi(I_j)\right) - C & \text{if} \quad \text{infeasible.} \end{cases} \tag{32}$$

where $J_{cost} = J\left(\Psi^S(I_j), \Psi^L(I_j), \Psi^{x_0}(I_j)\right)$.

Fitness evaluation consists of four steps: first, decode each individual to obtain all decision variables; second, use these decision variables to evaluate all EV state variables; third, evaluate cost and constraints; four, evaluate fitness according to feasibility.

5. RESULTS AND DISCUSSION

5.1. System Configuration

Three computational experiments were conducted. Each experiment varies in the number of customers and the fleet size, as detailed in Table 1. All experiments consider two CS; each of them with a limit of three parallel charging operations and the charging function in Figure 3. The charging function is an adaptation from the normal and slow charging functions provided by (Montoya et al., 2017).

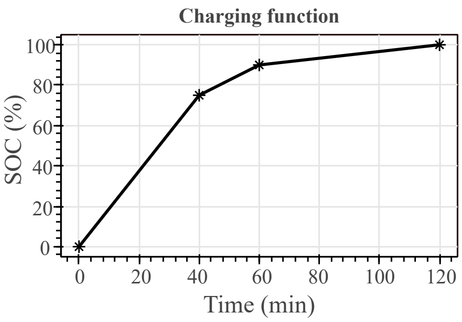


Figure 3. Charging function of each CS

Table 1. Experiments description

Exp. #	Network size	Customer nodes	Fleet size
Exp. 1	13	10	2
Exp. 2	38	35	3
Exp. 3	58	55	11

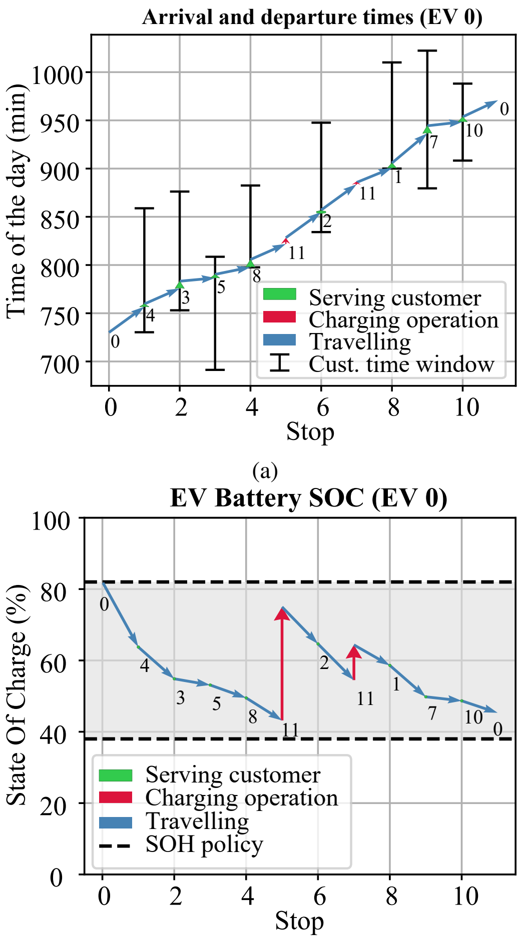
Instances are created by generating random nodes in a space of 20 km by 20 km. Travel times are the distance among all nodes multiplied by a factor of 1.7e-3, which is equivalent to EVs traveling at 35 km/hr. Similarly, per-weight energy consumption is obtained as the product of the distance among all nodes and the factor 2.8e-4. Customer time windows are generated from 9:00 hrs to 13:00 hrs with a random width within 1.5 hrs to 3.5 hrs.

The SOH policy is determined by $\alpha^- = 38\%$ and $\alpha^+ = 82\%$. Therefore, all EVs leave the depot with a SOC of 82%. An EV weight of 1.52 tonnes is considered. The maximum tour time duration is 360 minutes. The cost function weights are set to $(\omega_1, \omega_2, \omega_3) = (0.2, 0.8, 1.2)$. We choose these values to turn travel times and energy consumption into similar scales, while giving more importance to charging times and energy consumption.

The GA is implemented in Python 3.7.7 using the DEAP v1.3.0 library and runs in Windows 10 in an Intel Core i7-8750H @ 2.20GHz CPU with 16GB RAM. Each experiment has its set of GA hyper-parameters, which are detailed in Table 2.

Table 2. GA hyper-parameters per experiment

Hyperparameter	Exp. 1	Exp. 2	Exp. 3
Individuals	90	170	180
Generations	150	510	540
Crossover prob.	0.55	0.65	0.65
Mutation prob.	0.85	0.85	0.85
Penal. const.	5e5	5e5	5e5
Tournament size	3	3	3
n^*	2	3	3



(a)

(b)

Figure 4. Visualization of a feasible operation obtained from Experiment 1, case EV 0. In (a), the EV visits all customers within their time windows, even though two charging operations have been inserted. In (b), charging operations prevent EV SOC from operating outside SOC constraints.

5.2. Experiment Results

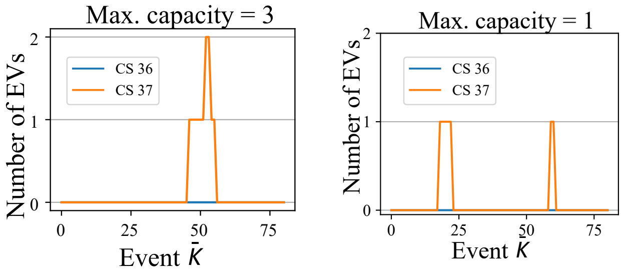
Table 3 summarizes the results of all experiments. All of them accomplish all constraints; thus, penalization is zero in all of them.

5.3. Contribution of Charging Operations to Feasibility

A proper insertion of charging operations allows EVs to: accomplish the SOH policy and accomplish time windows. The former result is evident, while the second one occurs because charging operations act as time-delay events. This time delays allow EVs to arrive at customers within their time windows. An example of this case is shown in Figure 4.a where, after visiting CS 11, the EV visits customer 2. If the charging operation is not inserted, then the EV arrives too early. Furthermore, Figure 4.b shows that the charging operation is required to fulfill the SOH policy.

Table 3. Experiments results

Experiment	Best fitness	Penalization	Total travel time	Total charging time	Total energy consumption	Max. CS occupation
Exp. 1	-201.85	0.0	273.34	57.53	97.68	1
Exp. 2	-225.98	0.0	336.83	30.19	152.97	2
Exp. 3	-511.56	0.0	768.05	18.33	419.93	1



(a) (b)

Figure 5. Number of EVs per CS. In (a), original result from Experiment 2. In (b), new result with less CS capacity.

5.4. Impact of Charging Station Capacities

Experiment 2 results give an operation where two vehicles visit a CS at the same time. This is not an issue because we have constrained the maximum capacity to three. However, if the maximum capacity is one, then the solution is infeasible. We repeat the experiment with CS capacities set to one, and increasing the number of generations to 650 and the population size to 200. The GA finds a feasible solution, with a maximum number of EVs of one. Figure 5 shows a comparison of the number of EVs per CS in both experiments.

The number of assigned customers influences the number of charging operations: the more customers to visit, the more likely to insert charging operations. Thus, it is not an issue if the fleet size is large because there is little chance for EVs to visit a CS and surpass the CS capacity. Experiments 2 and 3 show these cases: although the fleet size in Experiment 3 is larger than the fleet size in Experiment 2, the maximum occupation in Experiment 3 is lower than the maximum occupation in Experiment 2. This situation occurs because, in Experiment 2, EVs are assigned to 10 or more customers, while, in Experiment 3, EVs are assigned to five or six customers.

6. CONCLUSION AND FUTURE WORK

A Genetic Algorithm (GA) to solve the Electric Vehicle Routing Problem with Non-Linear Charging Functions, Battery Degradation, and Time Windows has been developed. We consider recharging infrastructure limitations, State of Health (SOH) policies to prevent early degradation of the Electric Vehicle (EV) batteries, and an EV model to accurately estimate travel times and energy consumption. Unlike other works, which focus in addressing a single or a small set of limitations, we tackle all limitations in a single formulation.

Our results show that the implemented GA is capable of finding feasible solutions in small and medium-sized problems. Feasible solutions ensure that CS capacities will never be exceeded, and the fleet will efficiently serve all customers. In addition, it has been shown that the GA can find routes when CS capacities are lowered. The resulting operation enables the fleet to minimize both short-term and long-term operational costs.

Future work involves developing a methodology to assign customers to EVs efficiently, in-depth analysis and improvement of genetic operators, and include traffic network stochasticity into the problem, specifically, into the EV state-space model. The latter will improve the prognosis of travel times and energy consumption.

ACKNOWLEDGMENT

This work has been partially supported by FONDECYT Chile Grant Nr. 1170044, the Advanced Center for Electrical and Electronic Engineering, AC3E, Basal Project FB0008, ANID, FONDEF Project ID18I10379, ANID, ANID PAI Convocatoria Nacional Subvención a Instalación en la Academia Convocatoria 2009 PAI77190021, and the Department of Electrical Engineering of the University of Chile.

NOMENCLATURE

S^i	Node sequence of EV i
L^i	SOC increase sequence of EV i
$x_1(0)^i$	Departure time of EV i
$x^i(k)$	State vector of EV i
$\theta(\bar{k})$	Counting vector at instant $\bar{k}$
D_j	Requirement at node j
K^0	Global counter range
K^i	EV i counter range
m	Fleet size
N^i	Customers vehicle i must visit
$\bar{P}_i$	Maximum weight of EV i
$\bar{Q}_i$	Battery capacity of vehicle i
r_j	Maximum capacity of CS j
T_j	Service time of customer j
T_j^+, T_j^-	Customer j time window bounds
$\bar{T}$	Maximum tour time

α^-,α^+	SOC safe zone
$F_j(\cdot)$	State transition function of state variable j
$\delta(\bar{k})$	Binary value tracking event kind at instant $\bar{k}$.
$\gamma(k)$	Vector tracking the event position at instant k.
t_{ij}	Travel time across arc ij
e_{ij}	Per-unit energy consumption across arc ij
Δ_i	Charging time at CS i

REFERENCES

Barco, J., Guerra, A., Muñoz, L., & Quijano, N. (2017). Optimal Routing and Scheduling of Charge for Electric Vehicles: A Case Study. *Math. Probl. Eng.*, *2017*, 8509783. doi: 10.1155/2017/8509783

Cattaruzza, D., Absi, N., Feillet, D., & González-Feliu, J. (2017). Vehicle routing problems for city logistics. *EURO J. Transp. Logist.*, *6*(1), 51–79. doi: 10.1007/s13676-014-0074-0

Dekker, R., Bloemhof, J., & Mallidis, I. (2012). Operations Research for green logistics - An overview of aspects, issues, contributions and challenges. *Eur. J. Oper. Res.*, *219*(3), 671–679. doi: 10.1016/j.ejor.2011.11.010

Erdelic, T., & Caric, T. (2019). A Survey on the Electric Vehicle Routing Problem : Variants and Solution Approaches. *J. Adv. Transp.*, *2019*, 1–48.

Faria, R., Marques, P., Moura, P., Freire, F., Delgado, J., & De Almeida, A. T. (2013). Impact of the electricity mix and use profile in the life-cycle assessment of electric vehicles. *Renew. Sustain. Energy Rev.*, *24*, 271–287. doi: 10.1016/j.rser.2013.03.063

Feng, W., & Figliozzi, M. (2013). An economic and technological analysis of the key factors affecting the competitiveness of electric commercial vehicles: A case study from the USA market. *Transp. Res. Part C Emerg. Technol.*, *26*, 135–145. doi: 10.1016/j.trc.2012.06.007

Froger, A., Mendoza, J. E., Jabali, O., & Laporte, G. (2017, jun). *A Matheuristic for the Electric Vehicle Routing Problem with Capacitated Charging Stations* [Research Report].

Gendreau, M., Potvin, J.-Y., Bräumlaysy, O., Hasle, G., & Løkketangen, A. (2008). Metaheuristics for the Vehicle Routing Problem and Its Extensions: A Categorized Bibliography BT - The Vehicle Routing Problem: Latest Advances and New Challenges. In B. Golden, S. Raghavan, & E. Wasil (Eds.), (pp. 143–169). Boston, MA: Springer US. doi: 10.1007/978-0-387-77778-8_7

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning* (1st ed.). USA: Addison-Wesley Longman Publishing Co., Inc.

Kumar, S. N., & Panneerselvam, R. (2012). A Survey on the Vehicle Routing Problem and Its Variants. *Intell. Inf. Manag.*, *04*(03), 66–74. doi: 10.4236/iim.2012.43010

Lee, R. J., & Sener, I. N. (2016). Transportation planning and quality of life: Where do they intersect? *Transp. Policy*, *48*, 146–155. doi: 10.1016/j.tranpol.2016.03.004

Montoya, A., Guéret, C., Mendoza, J. E., & Villegas, J. G. (2017). The electric vehicle routing problem with nonlinear charging function. *Transp. Res. Part B Methodol.*, *103*, 87–110. doi: 10.1016/j.trb.2017.02.004

Nazari, M., Oroojlooy, A., Snyder, L., & Takác, M. (2018). Reinforcement learning for solving the vehicle routing problem. In *Advances in neural information processing systems* (pp. 9839–9849).

Notter, D. A., Gauch, M., Widmer, R., Wäger, P., Stamp, A., Zah, R., & Althaus, H.-J. (2010, sep). Contribution of Li-Ion Batteries to the Environmental Impact of Electric Vehicles. *Environ. Sci. Technol.*, *44*(17), 6550–6556. doi: 10.1021/es903729a

Perez, A., Moreno, R., Moreira, R., Orchard, M., & Strbac, G. (2016). Effect of Battery Degradation on Multi-Service Portfolios of Energy Storage. *IEEE Trans. Sustain. Energy*, *7*(4), 1718–1729. doi: 10.1109/TSTE.2016.2589943

Pola, D. A., Navarrete, H. F., Orchard, M. E., Rabié, R. S., Cerda, M. A., Olivares, B. E., ... Pérez, A. (2015). Particle-filtering-based discharge time prognosis for lithium-ion batteries with a statistical characterization of use profiles. *IEEE Trans. Reliab.*, *64*(2), 710–720. doi: 10.1109/TR.2014.2385069

Sassi, O., Cherif, W. R., & Oulamara, A. (2014, oct). *Vehicle Routing Problem with Mixed fleet of conventional and heterogenous electric vehicles and time dependent charging costs.*

APPENDIX

A. METHODOLOGY TO CONSTRUCT θ VECTOR

Vector θ is built by following these directions:

1. Initialize a zero-vector of size $|V|$, and initialize $\bar{k} = 0$
2. Calculate times each EV i arrives at each node in its path S^i.
3. Calculate times each EV i leaves from each node in its path S^i.
4. Repeat $3 + 2 * \sum_{i=0}^{m}(s_i - 1)$ times the following: identify the next events according to arrival times and departure times. The event is characterized by the node where it occurs ($\gamma_j(\bar{k})$), and the event kind: arriving time or a departure time (i.e. $\delta(\bar{k})$). Do Eq. (7). Increase $\bar{k}$ by one.